



TITLE:

カオスの時系列の短期予測法の研究 (不確実性の下での数理モデルの構築と最適化)

AUTHOR(S):

木戸, 和彦; 古川, 長太

CITATION:

木戸, 和彦 ...[et al]. カオスの時系列の短期予測法の研究 (不確実性の下での数理モデルの構築と最適化). 数理解析研究所講究録 2001, 1194: 1-8

ISSUE DATE:

2001-03

URL:

<http://hdl.handle.net/2433/64826>

RIGHT:

カオスの時系列の短期予測法の研究

創価大学大学院 ○木戸和彦 (Kazuhiko Kido), 古川長太 (Nagata Furukawa)
Graduate school of engineering, Soka Univ.

1 はじめに

近年, カオスの研究が注目されている. その応用の 1 つにカオスの時系列の短期予測がある. カオスの振る舞いをする時系列は, 決定論に支配されているので, 確率的ではないという意味で予測可能であるといえる. しかし, その特徴の 1 つに「初期値に対して鋭く依存する」というものがあり, 短期間しか予測ができないということが知られている. その短期間の予測をする方法についての研究を行った.

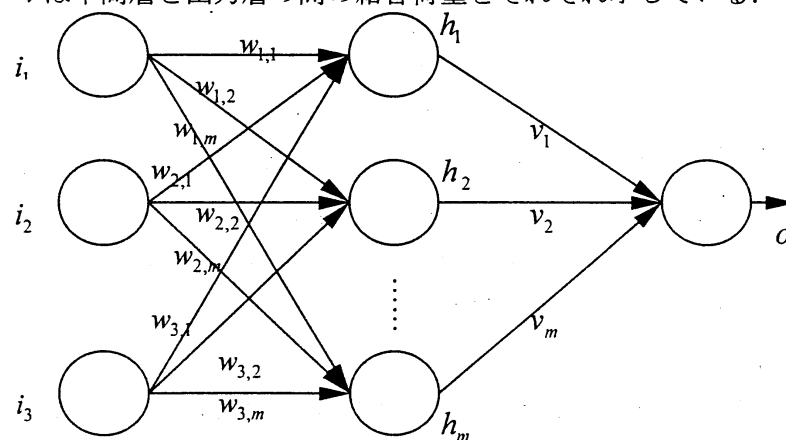
本研究は, 論文『ニューロコンピューティングを用いたカオスの時系列の短期予測』の中で提案した, 提案 1 による予測法の改良を行った.

2 ニューラルネットワークの構造

まず, 予測を行う際に用いるニューラルネットワークの構造を示す. このニューラルネットワークは, 入力層のユニット数を 3, 中間層のユニット数を m , 出力層のユニット数を 1 とした, 3 層からなる階層型神経回路網で構成されている.

論文『ニューロコンピューティングを用いたカオスの時系列の短期予測』の中の提案法との違いは, 中間層のユニット数を固定せずに, いろいろ変化させより良い結果が出たものを中間層のユニット数としたところにある.

図の i は入力層への入力, h は中間層からの出力, o は出力層からの出力, w は入力層と中間層の間の結合荷重, v は中間層と出力層の間の結合荷重をそれぞれ示している.



(図 1)

次に、従来の予測法の各層での入出力値の計算式を示した。ただし、関数 f はシグモイド関数と呼ばれるもので、1次元の場合は、式(4)であらわされる。また、 E は誤差関数を表し、 o が出力値で i が教師信号を表している。

入力層への入力値

$$i_k = x_k \quad (k: j, j+1, \dots, m+j-1, \quad j: 1, 2, \dots, 10-m) \quad \dots\dots\dots(1)$$

中間層からの出力値

$$h_k = f(\bar{h}_k)$$

$$\bar{h}_k = \sum_{l=j}^{m+j-1} i_l w_{kl} \quad (k, l: j, j+1, \dots, m+j-1, \quad j: 1, 2, \dots, 10-m) \quad \dots\dots\dots(2)$$

出力層からの出力値

$$o_k = f(\bar{o}_k)$$

$$\bar{o}_k = \sum_{l=j}^{m+j-1} h_l v_{kl} \quad (k: j, j+1, \dots, m+j-1, \quad j: 1, 2, \dots, 10-m) \quad \dots\dots\dots(3)$$

シグモイド関数

$$f(y) = \frac{1}{1 + e^{-y}} \quad \dots\dots\dots(4)$$

3. 従来のニューロコンピューティングを用いた時系列データの予測法

今、ある時系列データを x_i ($i=1, 2, \dots, 10$) とし、 x_i から x_{11} を予測する問題を考える。

従来の予測法では、まず次のパターンを先程のニューラルネットワークに学習させる。

	入力値	教師信号
パターン 1:	x_1, x_2, x_3	x_4
パターン 2:	x_2, x_3, x_4	x_5
\vdots	\vdots	\vdots
パターン 7:	x_7, x_8, x_9	x_{10}

ここで、パターン 7 まで全ての学習を終了した時点での結合荷重を保存しておく。そして予測過程として入力値を x_8, x_9, x_{10} 、結合荷重を今保存したものとし、同じニューラルネットワークで 1 度だけ計算させる。その結果を予測値 x_{11} とする。これが、従来の予測法である。

しかし、この従来の方法では、 x_{11} を予測する時点で、それに関連するはずの x_7 以前のデータの影響がほとんど無視されるという問題点がある。

そこで、以前、この問題を解決するために新しい方法を提案した。中でも提案 1 の方がより良い結果が得られたので、本研究ではこの提案 1 の方法に注目し、改良した。

4. タケンスの埋め込み定理

今、観測されたある時系列データ $y(t)$ から、ベクトル $(y(t), y(t-\tau), y(t-2\tau), \dots, y(t-(n-1)\tau))$ をつくる (τ は遅れ時間). このベクトルは n 次元再構成状態空間 R^n の 1 点を示すことになる. 従って t を変化させると、この n 次元再構成状態空間に軌道が描ける. もし、対象のシステムが決定論的力学系であって、観測時系列データがこの力学系の状態空間から 1 次元ユークリッド空間 R への C^1 連続写像に対応した観測系を介して得られたものと仮定すれば、この再構成軌道は元の決定論力学系の埋め込みになっている. つまり、元の力学系に何らかのアトラクタが現れているならば、再構成状態空間にはこのアトラクタの位相構造を保存したアトラクタが再現することになる.

5. 提案 1 と提案 1・改の予測法

提案 1 または提案 1・改 (どちらも共通) の予測法では、まず、元の時系列データにタケンスの埋め込み定理を適用し、3 次元再構成状態空間に埋め込みを行う. すると、元の時系列データは、 (x_{i-2}, x_{i-1}, x_i) ($i=3, 4, \dots, n$) と書き直すことができる.

ここで、最初のニューラルネットワークに学習させる. この場合入力データが 3 次元なので、それぞれの層への入出力値の計算式は、次のようになる.

提案 1 の場合

入力層への入力値

$$i_k = (x_k, x_{k+1}, x_{k+2}) \quad (k: j, j+1, \dots, m+j-1, \quad j: 1, 2, \dots, 8-m) \quad \dots\dots\dots(5)$$

中間層からの出力値

$$h_k = (f(\bar{h}_k), f(\bar{h}_{k+1}), f(\bar{h}_{k+2}))$$

$$\bar{h}_k = \sum_{l=j}^{m+j-1} i_k w_{kl} \quad (k, l: j, j+1, \dots, m+j-1, \quad j: 1, 2, \dots, 8-m) \quad \dots\dots\dots(6)$$

出力層からの出力値

$$o_k = (o_1, o_2, o_3) = (f(\bar{o}_k), f(\bar{o}_{k+1}), f(\bar{o}_{k+2}))$$

$$\bar{o}_k = \sum_{l=j}^{m+j-1} h_k v_{kl} \quad (k: j, j+1, \dots, m+j-1, \quad j: 1, 2, \dots, 8-m) \quad \dots\dots\dots(7)$$

シグモイド関数

$$f(y_1, y_2, y_3) = (\bar{f}(y_1), \bar{f}(y_2), \bar{f}(y_3))$$

$$\bar{f}(y) = \frac{1}{1 + e^{-y}} \quad \dots\dots\dots(8)$$

提案 1・改の場合

入力層への入力値

$$i_k = (x_k, x_{k+1}, x_{k+2}) \quad (k: j, j+1, \dots, m+j-1, j: 1, 2, \dots, 8-m) \quad \dots\dots\dots(9)$$

中間層からの出力値

$$h_k = (f(\bar{h}_k), f(\bar{h}_{k+1}), f(\bar{h}_{k+2}))$$

$$\bar{h}_k = \sum_{l=j}^{m+j-1} i_k w_{kl}$$

$$(k: j, j+1, \dots, m+j-1, l: j, j+1, j+2, \dots, m, j: 1, 2, \dots, 8-m) \quad \dots\dots\dots(10)$$

出力層からの出力値

$$o_k = (o_1, o_2, o_3) = (f(\bar{o}_k), f(\bar{o}_{k+1}), f(\bar{o}_{k+2}))$$

$$\bar{o}_k = \sum_{k=j}^{m+j-1} h_k v_k \quad (k: j, j+1, \dots, m+j-1, j: 1, 2, \dots, 8-m) \quad \dots\dots\dots(11)$$

シグモイド関数

$$f(y_1, y_2, y_3) = (\bar{f}(y_1), \bar{f}(y_2), \bar{f}(y_3))$$

$$\bar{f}(y) = \frac{1}{1 + e^{-y}} \quad \dots\dots\dots(12)$$

また、この場合、データが 3 次元なので、誤差関数はそれぞれ E_1 , E_2 , E_3 と 3 つ求められる。そこで、 μ と λ を使ってそれぞれに比率を持たせ、誤差関数 E と定義する。

$$E = \lambda E_1 + \mu E_2 + (1 - \lambda - \mu) E_3$$

$$(0 < \lambda < 1, 0 < \mu < 1)$$

ただし、数値計算をする時は μ , λ を共に 0.1 きざみで $(\mu + \lambda) < 1$ という条件が成り立つ全通りを計算し、最も良いものを採用する。

この計算式を使い、最初のニューラルネットワークに次のパターンを学習させる。

入力値	教師信号
パターン 1: $(x_1, x_2, x_3), (x_2, x_3, x_4), (x_3, x_4, x_5)$	(x_4, x_5, x_6)
パターン 2: $(x_2, x_3, x_4), (x_3, x_4, x_5), (x_4, x_5, x_6)$	(x_5, x_6, x_7)
\vdots	\vdots
パターン 5: $(x_5, x_6, x_7), (x_6, x_7, x_8), (x_7, x_8, x_9)$	(x_8, x_9, x_{10})

この作業を 10 個のデータが全て無くなるまで繰返し行う。データが無くなった時点の結合荷重を保存しておく。そして予測過程として入力値を $(x_6, x_7, x_8), (x_7, x_8, x_9), (x_8, x_9, x_{10})$, 結合荷重を

今保存したものとし、同じニューラルネットワークで1度だけ計算させる。その結果の内、1番最後のものを予測値 x_{11} とする。

次に、埋め込み次元を上げていき、7次元に埋め込みする。すると元の時系列データは、 $(x_{i-6}, x_{i-5}, \dots, x_i)$ ($i = 7, 8, 9, 10$) と書き直すことができる。そして、3次元の場合と同様に、次のパターンを学習させ、終了した時点での結合荷重を保存しておく。

$$\begin{array}{ll} \text{入力値} & \text{教師信号} \\ \text{パターン: } (x_1, \dots, x_7), (x_2, \dots, x_8), (x_3, \dots, x_9) & (x_4, \dots, x_{10}) \end{array}$$

次に、予測過程として入力値を $(x_2, \dots, x_8), (x_3, \dots, x_9), (x_4, \dots, x_{10})$ 、結合荷重を先程保存したものとして、1度だけ計算する。その結果の内、1番最後のものを予測値 x_{11} とする。

以上が提案1、提案1・改の予測法であるが、この方法だと、 x_{11} を予測する時、初期値以外の全てのデータの影響が繁栄され、従来の方法での問題が解決できると思われる。

6. 従来の予測法と提案1または提案1・改との相違点

従来の予測法と提案1または提案1・改との相違点は、

- 中間層からの出力値と出力層からの出力値が3次元になる点
- 誤差関数が3つ求まる点

つまり、従来の予測法は1次元入力1次元出力、それに対し、提案1または提案1・改は多次元入力多次元出力、という点が大きく異なる。

7. 数値実験とその結果

次に、提案1と提案1・改の有効性を確認するために代表的なカオス的振る舞いをする時系列モデルの短期予測に適用し、その結果を示す。

まず、時系列モデルを式(13)とする。

$$x_{n+1} = ax_n(1-x_n) \quad (0 \leq a \leq 4, 0 \leq x_0 \leq 1) \quad \dots\dots\dots(13)$$

集積点 $\bar{a} = 3.5699\dots$

これは、ロジスティック写像と呼ばれるもので、パラメータ a の値が集積点 \bar{a} をこえると、カオス的振る舞いをするということが知られている。

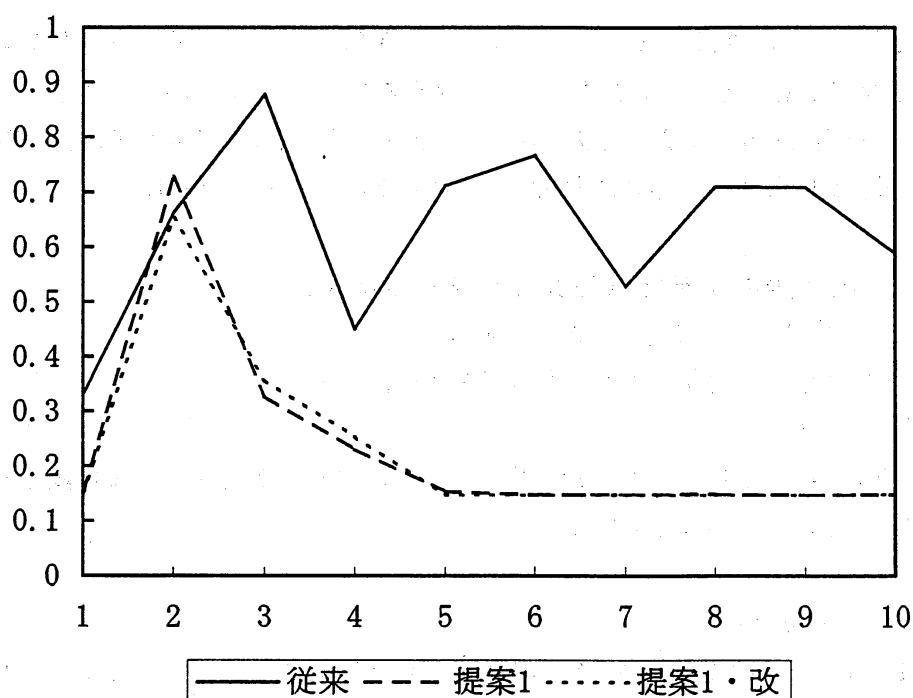
以下に示す表は、モデルから発生させた時系列データを従来の予測法と提案1と提案1・改で予測した結果を示したものである。

ステップ数	1	2	3	4	5
実際の値	0.147865	0.504004	0.999936	0.000257	0.001026
従来	0.329725	0.662155	0.878268	0.450339	0.711553
提案1	0.149937	0.728111	0.327070	0.230424	0.153293
提案1・改	0.148751	0.653076	0.356112	0.253599	0.146962

6	7	8	9	10
0.004099	0.016329	0.064250	0.240488	0.730614
0.766417	0.528588	0.710065	0.708902	0.588553
0.147559	0.148425	0.148889	0.147050	0.147716
0.147014	0.147765	0.147387	0.146953	0.148392

(表 1)

次に、この表をグラフにしたものを示す。図は、横軸に時間ステップ、縦軸に（実測値－予測値）をプロットしたものである。



(図 2)

この表や図から分かるように、提案 1・改の予測が最も良い値を多く示している。

8. 応用

以上の結果から提案 1、提案 1・改の予測法が有効であることが確認できた。そこで、応用として、日経平均株価に適用した。

まず、数値結果を表 2 に示す。

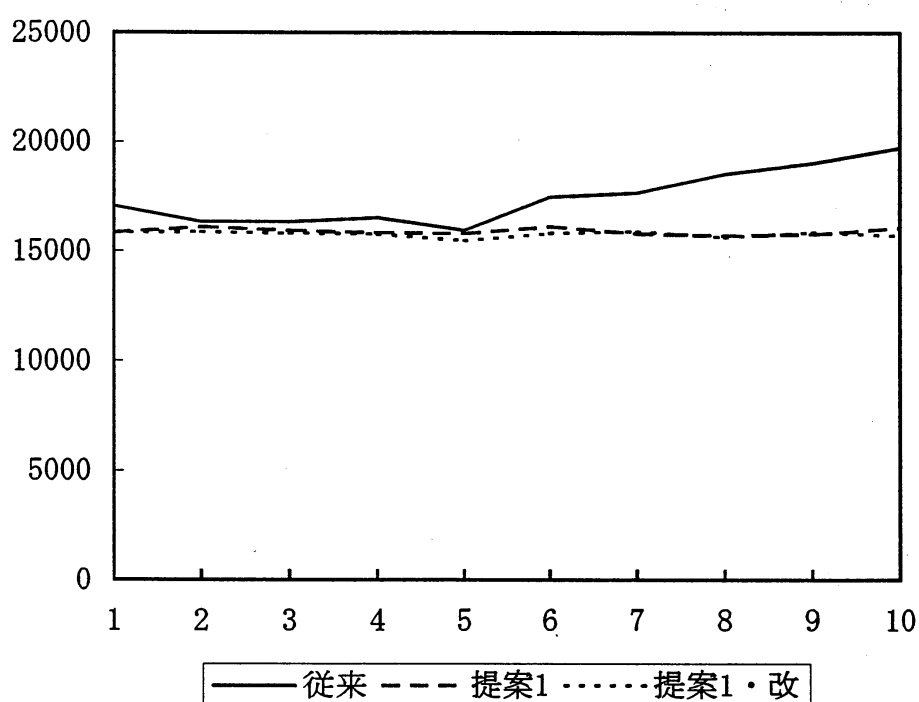
ステップ数	1	2	3	4	5
実際の値	15803.2	15881.0	16499.1	16295.4	15238.1
従来	17053.5	16339.8	16339.2	16534.9	15951.7
提案 1	15847.1	16100.4	15946.2	15839.4	15816.0
提案 1・改	15846.2	15871.6	15806.5	15778.1	15483.3

6	7	8	9	10
14955.6	14790.4	15259.0	15086.0	14588.4
17482.6	17666.3	18544.1	19032.8	19720.8
16138.0	15818.4	15728.1	15784.8	16070.5
15822.3	15921.3	15647.7	15872.7	15706.5

(単位：円)

(表 2)

次に、この表をグラフにしたものを示す。図は、横軸に時間ステップ、縦軸に | 実際の株価 - 予測した株価 | をプロットしたものである。



(図 3)

9. 最後に

以上のような結果から、従来の予測法より提案 1, 提案 1 より提案 1・改の方がより良い結果が得られることがわかった。これは、入出力を多次元化したこと、また、中間層を変動可能にしたことによるものだと考えられる。また、ロジスティック写像の予測結果から提案した予測法の有効性が確認できた。

最後に、今後この研究は、株価の予測の例のように、カオス的時系列データの短期予測に役立って行くと思われる。

<参考文献>

1. 五百旗頭正, 菅家正康, 藤本泰成, 鈴木新伍: “カオス的時系列の短期予測のための局所ファ

- ジィ再構成法”，日本ファジィ学会誌 vol1.7, No1, pp.186-194
2. 合原一幸編著：“ニューロ・ファジィ・カオス” 新世代アナログコンピューティング入門
 3. 合原一幸編著：“カオス “カオス理論の基礎と応用
 4. D.E.Rumelhart, G.E.Hinton and R.J.Williams : Nature, 323, pp.533-536(1986)
 5. 木戸和彦, 菊池洋一：“ニューロコンピューティングを用いたカオスのデータの短期予測”，数理解析研究所講究録 1043 決定理論とその関連分野, pp.50-58(1998)
 6. 木戸和彦：“ニューロコンピューティングを用いたカオスの時系列の短期予測”，第 14 回ファジィ・システム・シンポジウム講演論文集, pp.763-764(1998)
 7. Kazuhiko Kido : STOCHASTIC MODELS IN ENGINEERING, TECHNOLOGY AND MANAGEMENT, pp.278-284(1999)